STINFO COPY

AFRL-HE-WP-TR-2007-0068

Testing Adaptive Levels of Automation (ALOA) for UAV Supervisory Control

Rubin Johnson
Michael Leen
Daniel Goldberg

OR Concepts Applied

7032 Comstock Avenue, Suite 100
Whittier CA 90602

March 2007

Final Report for March 2005 to March 2007

Air Force Research Laboratory
Human Effectiveness Directorate
Warfighter Interface Division
System Control Interfaces
Wright Patterson AFB OH 45433

NOTICE AND SIGNATURE PAGE

TECHNICAL REVIEW AND APPROVAL

AFRL-HE-WP-TR-2007-0068

THIS TECHNICAL REPORT HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION.

FOR THE DIRECTOR

//signed//
DANIEL G. GODDARD
Chief, Warfighter Interface Division
Air Force Research Laboratory

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
24 Mar 2007

2. REPORT TYPE
Final

3. DATES COVERED *(From - To)*
25 Mar 2005 to 24 Mar 2007

4. TITLE AND SUBTITLE
Testing Adaptive Levels of Automation (ALOA) for UAV Supervisory Control

5a. CONTRACT NUMBER
FA8650-05-C-6547

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
65502F

6. AUTHOR(S)
Rubin Johnson, Michael Leen, Daniel Goldberg

5d. PROJECT NUMBER
3005

5e. TASK NUMBER
HC

5e. WORK UNIT NUMBER
59

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
OR Concepts Applied
7032 Comstock Ave., Ste. 100
Whittier CA 90602

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Materiel Command
Air Force Research Laboratory
Human Effectiveness Directorate
Warfighter Interface Division
System Control Interfaces Branch
Wright-Patterson AFB OH 45433-7511

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/HECI

11. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-HE-WP-TR-2007-0068

12. DISTRIBUTION AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES
AFRL/PA Cleared 7/17/07 AFRL/WS-07-1657.

Report developed under Small Business Innovation Research (SBIR) Phase II Contract for Topic # AF04-071.

14. ABSTRACT
OR Concepts Applied has designed and implemented a human factors test bed to evaluate adaptive autonomy schemes and a range of levels of autonomy for UAV supervisory control. The test bed includes a high fidelity simulation that interfaces with a mission control element (MCE) that supports multi-vehicle control. The MCE uses OPUS mission planning tools to provide the operator with optimization and analysis decision aids. SA Technologies, Inc. was instrumental in helping to create user interface elements to help maintain high levels of situation awareness even when the level of automation is increased. The test bed provides a variety of tools for researchers to create scenarios, alter adaptive schemas, and collect experimental data.

15. SUBJECT TERMS
UAV, span of control, OPUS, mission planning, adaptive autonomy, situation awareness

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
UNCL	UNCL	UNCL	SAR	32	Gloria Calhoun
					19b. TELEPONE NUMBER (*Include area code*)

Standard Form 298 (Rev. 8-98)

Prescribed by ANSI-Std Z39-18

THIS PAGE INTENTIONALLY LEFT BLANK.

Table of Contents

List of Figures

1 Introduction

1.1 Overview

This is the final report for the Phase II Small Business Innovation Research (SBIR) effort entitled *Adaptive Levels of Automation (ALOA) for UAV Supervisory Control* conducted by OR Concepts Applied (ORCA) for the Air Force Research Laboratory (AFRL). Our research is directed towards increasing spans of control so that one person can effectively control multiple unmanned air vehicles (UAV) from a map-based mission control station. The goal of the ALOA effort was to devise and implement a test bed to evaluate schemes for adaptive levels of autonomy for UAV supervisory control. The tool produced in this effort provides an environment to test level of automation control strategies and the role of the humans in optimizing system performance. Our work explores the notion of a mission control element with enough logic to change the level of automation to keep the human's workload manageable while maintaining situation awareness (SA). Increasing the level of automation can lead to an "out-of-the-loop" phenomenon. Maintaining SA is a key aspect of effective supervisory control. The ALOA test bed provides an environment to test mission planning components as well as situation awareness tools that will help increase the span of control.

The high fidelity representation of the underlying mission planning problems and the sophistication of the analysis and optimization tools are two features that make the ALOA test bed stand out. ORCA's expertise in operational mission planning and unmanned systems provided the basis for a functional mission control element. The primary tasks of aircraft planning are accurate representations of the cognitive load for actual multiple UAV operations in a military environment. Goal directed task analysis was used to improve the user interface. Other important innovations include: novel presentations of sortie routes, mission events, and risk levels; design and implementation of relevant automation levels for a variety of tasks; and specialized researcher tools. Not only can the researcher use a script editor tool to create scenarios, experimental results are logged to an XML file that can be interpreted by a parser that is part of the test bed.

In the ALOA effort, ORCA has made great strides in both mission control element (MCE) design and the implementation of levels of autonomy and the means to adapt the autonomy levels dynamically. Working with human factors experts at AFRL and SA Technologies, ORCA has refined MCE elements to provide an effective human-system interface (HSI) design for the MCE. Mission planning tools, provided through the ORCA Planning and Utility System (OPUS), include state of the art allocation and route planning tools that provide a realistic planning environment. Tools for the researcher allow a wide range of experimental scenarios to be designed. Tools are included to assess situation awareness, workload and performance, and to record the results of experimental runs.

1.2 Background

1.2.1 Multi-UAV Control

Although the focus of this effort is to design a test bed and associated tools that will be used to experiment with autonomy concepts for UAV supervisory control, it is important to keep in mind the context of the problem and the larger goal: enabling multi-vehicle supervisory control. Multi-

UAV supervisory control refers to a control concept in which a single person manages a pod of UAVs. In this concept, UAV flight control is autonomous and the human participates in planning, problem solving, and contingency operations (for example, a system failure). Several unmanned vehicle programs envision a future in which unmanned vehicles work together in teams and are controlled by a single person acting in a supervisory role. The J-UCAS concept involves a single pilot controlling a group of four Unmanned Combat Air Vehicles (UCAVs). The Air Force plans to use teams of Predators and armed Predator Bs to perform hunter-killer missions. The Office of the Secretary of Defense UAV Roadmap (December 2002) calls for improvements in multi-vehicle supervisory control capabilities.

It may seem like a small matter but controversy is attached to the designations of the people at the controls of the UAV. In the early days when all UAVs were remotely piloted vehicles, it was straightforward to use the term pilot. The knowledge that pilots have of aerodynamics and airplane operations were crucial when MCEs were essentially flight decks on the ground. As technology has improved, it is possible to manage an aircraft's flight path using waypoint control. The user constructs a route plan using software that prevents infeasible flight plans from ever being generated. The waypoints are then communicated to the aircraft which flies the path using onboard autopilots. With such technology, a person with less training than a pilot can effectively manage unmanned aircraft. Other operators may have special training for managing UAV payload and interpreting imagery and intelligence data that may be gathered. In this report, the term most frequently used will be operator in keeping with the notion that automated tools will be available to maintain flight feasibility. The operator will still be kept in the loop to employ human judgment where appropriate. We are not precluding the operator from also being a trained pilot although we expect that it will not be a requirement for every service.

Increasing the capability of C2 decision aids and situation awareness tools, and implementing autonomous execution of tasks (such as target allocation and route planning) will help increase the span of control; however, more research and experimentation is required to determine the best use of these methods and tools. The current situation falls short of the goal of multi-vehicle supervisory control. While autonomous flight control is possible because it is more tractable, true multi-vehicle control is still in its infancy. Some current unmanned vehicle systems require more than one person to control a single vehicle. For example, the Navy's Tactical Control System (TCS) currently requires two operators to control a single UAV and its sensors: a pilot controls and monitors the health and location of the vehicle, and the sensor operator manages the sensor payload and the data that is being transmitted back to the control station via the vehicle's communications system.

It is clear that if a single operator is going to control a group of UAVs, some tasks will have to be automated to some degree. While autonomous operations will play an important role in achieving multi-vehicle control, the human factor is critical. One obvious role for the human is to intervene in case of system failure. Another important role is for the operator to intervene when automated tools fail because of invalid modeling assumptions or algorithmic idiosyncrasies. Automated mission planning tools use underlying models of the real world and algorithms to solve problems. On rare occasions, the solutions will be suboptimal due to invalid underlying assumptions. Automated tools may also produce poor results because of bad data. In such cases, the operator must intervene to modify the answer. Making use of human experience and knowledge is an important aspect of optimizing multi-vehicle control system performance.

To allow the operator to perform planning, monitoring, and intervention duties effectively, the system must provide the operator with situation awareness and manage the operator's workload (or permit the operator to manage the workload). Situation awareness requires data, but providing too much data, or data that is difficult to understand, will diminish situation awareness. To enhance situation awareness, the right data must be provided to the operator when needed and in a form that is easily understandable. Exploring tools that enhance situation awareness and performance is another important dimension of this effort

1.2.2 Multi-Vehicle Mission Planning Capabilities

Mission planning is decision making to address air war force employment. The basic problem is to avoid threats and accomplish mission objectives. There are several aspects of mission planning for groups of unmanned vehicles, including task allocation, route planning, data collection requirements, communications planning, dynamic replanning, and multi-vehicle coordinated and cooperative planning.

Allocation determines which vehicle will perform which mission tasks. Route planning determines the path the vehicle will follow and may need to take into account factors such as the vehicle's tasks, terrain, restricted areas and no-fly zones, vehicle performance, environmental factors such as weather or ocean currents, multi-spectral signature information, threats, and payload capabilities/imaging quality requirements. Data collection planning includes sensor control and managing imaging requirements. Communications planning deals with how and when to transmit data and takes into account issues such as potential line of sight link locations, satellite availability, and communications frequencies.

Dynamic replanning involves replanning the mission after vehicles are underway. Replanning may be triggered by a wide range of factors, including new threats or targets, changes in no-fly zones or rules of engagement, new mission tasks, new intelligence or Battle Damage Assessment (BDA) data, changes in the health and status of a vehicle, or loss of communications. The time frame to react to changes will dictate the type of replanning that is possible. For example, if a vehicle must react in seconds to avoid a threat, then an evasive maneuver may have to be executed, possibly followed by a replanning of the vehicle's mission. If the time frame is longer, the first step in the replanning process is to analyze the change in mission quality and effectiveness because of the change in planning data. For example, if a new threat is detected but has little impact on route quality, then it may not be necessary to replan. Once the new planning data is analyzed for the impact on the current plan, replanning can be performed as needed.

Multi-vehicle coordinated and cooperative planning enables teams of UAVs to avoid conflicts and to accomplish missions that require teamwork. Task allocation must take into account cooperative behavior required to accomplish a task, such as multiple sensor looks required to identify a vehicle. Coordinating route planning includes assigning ingress/egress paths to vehicles, making sure that vehicles maintain safe distances from each other, and invoking other measures to deconflict routes, such as designating areas of operation for each vehicle or assigning set altitudes to each UAV.

Allocation is a rich set of problems with wide scope and a varied nature. A key parameter is the size of the problem in terms of numbers of assets and tasks. Heterogeneous asset problems tend to be more difficult than if all assets are alike. Synchronization constraints add to the difficulty. In military operations, the air tasking order represents a solution to a daily allocation problem

that is faced by air commanders. In ALOA, we focus on a smaller scale allocation problem in which a set of imaging and strike tasks must be assigned to a pod of 2-8 UAVs. Unfortunately, even small problems can be quite complicated, especially if one seeks an optimal solution.

1.3 Phase II Technical Objectives

There were four technical objectives to accomplish in this Phase II effort. In this section, we describe the goals and our accomplishments relative to those goals.

1. Refine the levels of autonomy defined in Phase I for allocation, route planning, imagery analysis, and weapon control.
2. Mature the ALOA architecture.
3. Finalize the mission control station emulator (MCSE) design and provide a functional research test bed.
4. Evaluate the ALOA architecture in a representative high-fidelity UAV simulation environment.

The many possibilities for levels of autonomy (LOA) caused this task to expand more than originally considered. Since there are many tasks associated with UAV control, it is possible to have a variety of autonomy levels, not just a single system level autonomy. Our final design accommodates both ideas by allowing the notion of a set of system level autonomy levels, each of which maps to a unique set of task level autonomy settings. Moreover, the system can be set to run just the system levels or allow the user to change individual task LOA settings. Another complication arises in that it may be reasonable to have different levels of autonomy associated with individual sorties. That is also an option that may be permitted. Although we have created a system with considerable flexibility, great pains were taken to eliminate levels of autonomy that seemed unsuitable or irrelevant for particular tasks. Sections 3.1 and 3.2 cover this in detail.

Setting up an architecture that would allow different schemes for altering methods of changing the levels of autonomy is perhaps the most important part of this project. We were not trying to find the best scheme for changing levels of autonomy; we are building a test bed to allow researchers to experiment with various schemes for LOA changing. Our program allows the operator to change the levels of autonomy (adaptable), or the system to be responsible for such changes (adaptive). There can be mission-phase based and contingency-specific adaptation of LOAs. There is the capability to adjust LOAs based on operator workload or performance.

At the midterm, there was a functional test bed that was robust enough to be used by over 80 attendees at the Orlando 2006 Association of Unmanned Vehicle Systems International (AUVSI) trade show. The system was stable enough to be used by novices. See Figure 1. Given only 15 minutes of instruction, users were able to control multiple UAVs in a simulation environment with varying degrees of success. The key is that these novices only needed to know enough to control the software which in turn understands the vehicles enough to keep them flying. The operators could focus on the strategic and tactical employment of the UAVs. The final version of the software has additional features that more adequately capture cognitive tasks that will confront modern day UAV operators.

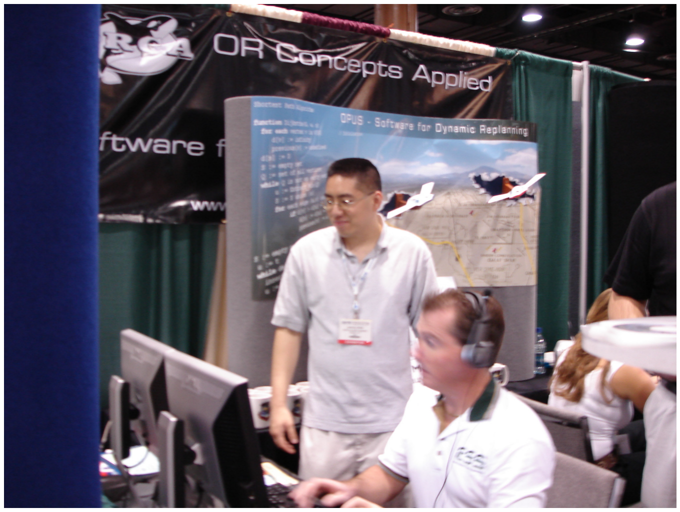

Figure 1 ALOA Demo at AUVSI

The OPUS simulation environment provides users of the ALOA test bed with a rich experience representative of a realistic UAV mission. There are representative sets of tasks that must be allocated to the vehicles. System failures create a need to dynamically reallocate. Such reallocations are but one reason that may necessitate a rerouting of sortie paths. Threats and targets may appear. Vehicles have realistic performance as do threats. We look forward to the exploitation of the ALOA tool in a number of experiments.

2 Mission Control Element

One of the most important outcomes of our research was establishing our thoughts about the role of the software-based mission control element as part of a system of people and UAVs. Based on interviews with UAV pilots and operators, demonstrations of the Navy's Tactical Control Station, participation in the J-UCAS program (including the successful flight of two cooperating X-45s in August 2005), it seems obvious that the role of the mission control station or mission control element (MCE) is to enable the management of a pod of unmanned aircraft. The MCE is used to plan – control – and analyze UAV employment. The human at the MCE must maintain a high level of situation awareness and vigilance to supervise the UAVs.

2.1 ALOA MCE Requirements

While working on this project, we needed to focus on the tasks that confront the operator and how the MCE facilitates those tasks. To that end, we wanted to insure that our mission control station emulator (MCSE) had the following capabilities.

I. Alerts the user to changes in
   1. Mission
   2. Vehicle health and status
   3. Environment
      a. Threat

b. Friendly – rules of engagement, airspace control mechanisms (keep out zones, refueling tracks, etc.)

c. Nature – weather, wind, etc.

II. Helps the operator address that ever looming question – What do I do now? Tools for allocation, autorouting, weapon employment, deconfliction, etc., are important.

III. Helps the user answer five types of questions:

1. How is it going?
2. Still on track?
3. What if?
4. What happened [especially if something goes wrong]?
5. Why did it [autonomous activities] do that [even if good]?

In our interviews with UAV controllers, we found that these five types of questions captured the types of questions that they had. We also found that several operators wanted more and more vehicle health and status information (e.g. oil pressure and temperature). When pressed as to why this information was useful, they replied that they were simply curious and that it wasn't that useful in completing their missions.

2.2 ALOA MCE Features

The ALOA MCSE does indeed help the user accomplish tasks. Situation awareness of the mission, vehicle health and status, and the environment are addressed by various human system interface (HSI) elements.

1. Chat window presents rules of engagement (ROE) and mission updates
2. A scrolling ticker provides various warnings and system updates
3. Health and Status Indicators change color
4. Map based displays show the environment
5. Pop Up Threat Indicators (visual and aural)

Planning tools help the operator answer what to do next. If there is a new threat, metrics help the user decide what the impact is. The SAM shot evasion mechanism emulates a tactical fast threat avoidance mechanism. If a new route is needed, tools help generate the route. If a system failure causes a sortie to be unable to complete its mission objectives, the operator has tools to reallocate tasks. Figures of merit (FOM) – force level and sortie level – provide information that can be used to guide decision making.

The same FOMs that help in making decisions also help the operator answer questions about how things are going. Comparing initial FOMs to current FOMS are precisely what helps one compare the current situation with original expectations. The tools provided in the ALOA environment also give the operator the ability to change plans and reassess WITHOUT

committing to the new plan. This lets the operator answer "what if" questions. The ability to use the traversal tool to project the aircraft into the future is another aspect of this capability.

Tools for answering "what happened?" and "why did the system do that?" are more subtle. One can review logs to see what happened as well as traverse the past. The ability to show low probability events that happened to occur can be quite useful. Being able to inspect algorithm parameters can provide some insights into automated solutions, however, it must be understood that automated route planning algorithms can produce good results that are still counter-intuitive.

3 Adaptive Levels of Autonomy

For systems that involve human-computer interaction, alternative automation schemes have been proposed in human factors research to address the problems associated with approaches that automate system tasks and leave the human in a role as monitor. One scheme is a human-centered approach to automation, in which the human and the machine work together as a system. The goal is to keep the human in the control loop by having the human perform meaningful tasks. Two complementary human-centered approaches have been developed that address the performance problems associated with automation: Levels of Autonomy and Adaptive Autonomy.

The ALOA software was designed to provide a human factors research test bed to evaluate adaptive autonomy and a range of levels of autonomy for UAV supervisory control. In this section, we describe levels of autonomy (LOA) implemented in ALOA and several schemes for adapting the LOA. We begin with some background, which includes Parasuraman's ten-level hierarchy of LOA [1]. The next section describes the LOA implemented in ALOA for four operator tasks: weapon release authorization, image analysis, allocation, and autorouting. Finally, we discuss system mechanisms for changing the LOA.

3.1 Levels of Autonomy

A level of autonomy (LOA) refers to a distribution of workload between the human and the computer. At one extreme (*manual*), the human makes all decisions and performs all actions; at the other extreme (*fully autonomous*), the computer acts autonomously. For the intermediate levels, control tasks are divided between the computer and the human to optimize human and system performance. This approach allows the human to stay in the loop, but doesn't require that every task be performed manually.

Within a system in which multiple functions must be performed, different functions can have different autonomy levels. Consider the example of controlling a UAV. A human operator could be responsible for interpreting sensor images, rather than using an ATR system, but an automated control system could be responsible for flying the aircraft, including generating the route and rerouting the aircraft in case of changes in the threat lay-down or other environmental changes. One task is performed entirely by a human, the other by the computerized flight control system. At an intermediate level, if a new threat pops-up, the computer could generate a new route plan along with route quality metrics for the current route and the new route and allow the operator to select from the two options.

While there are a continuum of possible autonomy levels between manual and fully autonomous, in practice, various systems have been implemented with ten or fewer levels. See [1] and [2] for

examples of LOA hierarchies. In [1], Parasuraman (et al.) gives the following ten levels of autonomy:

Table 1 Parasuraman Levels of Autonomy

Level	Description	Type
10	The computer decides everything, acts autonomously, ignores the human	Fully Automatic
9	Informs the human only if the computer decides to	
8	Informs the human only if asked, or	
7	Executes autonomously, then necessarily informs the human, and	Automatic with feedback
6	Allows the human a restricted time to veto before automatic execution,	Veto
5	Executes that suggestion if the human approves, or	Consent
4	Suggests one alternative	
3	Narrows the selection down to a few, or	Multiple Options
2	Offers a complete set of decision/action alternatives, or	
1	Offers no assistance; human must take all decisions and action	Manual

Parasuraman notes that this LOA scale refers to output functions performed by the system: decision and action selection. Automation can also be applied to input functions: acquiring and processing information. To expand the model, Parasuraman adds a simple four-stage model of human information processing. From the human information processing model, four classes of system input and output functions are given to which automation can be applied: (1) Information acquisition, (2) Information analysis, (3) Decision and action selection, and (4) Action implementation. Each of these functions can be automated at any one of the ten autonomy levels.

As an example of the use of this multi-stage model of automation, Parasuraman applies the model to make suggestions about the automation of air traffic control (ATC) systems. ATC systems are being redesigned because the volume of air traffic is projected to double over the next several years and many system tasks will need to be automated to lessen the burden on air traffic controllers. Parasuraman recommends that information acquisition and analysis can be automated at high levels, provided the system is proven reliable. However, decision and action selection and implementation should only be automated at high levels for low-risk situations. For high-risk situations, the automation should be set at a much lower level with the computer suggesting alternatives to the controller, who chooses and executes one of the actions.

3.2 ALOA LOA

This section describes the levels of autonomy used in the final Phase II version of the ALOA software. ALOA has LOA hierarchies for four operator tasks: weapon release authorization, image analysis, allocation, and autorouting.

3.2.1 Weapon Release Authorization

For the weapon release authorization task, an operator must examine an image that depicts the designated point of impact (DMPI) of an upcoming weapon release task. The operator must then authorize the weapon release against this target in a timely fashion. The time limit insures that authorization occurs before the aircraft reaches its weapon release point. In ALOA, the operator should answer a yes/no question that would indicate whether the weapon should be released.

1. Manual: a yes/no question is asked; the operator must decide if the weapon release should be authorized.
2. Consent: a yes/no question is asked, but an answer is pre-selected, which represents the computer's suggestion; the operator must choose one of the two options before the decision time expires; if time expires on this task, however, no action is taken and the weapon release is not authorized.
3. Veto: a yes/no question is provided and an answer is pre-selected, which represents the computer's suggestion; the operator must choose one of the two options before the decision time expires; if the user takes no action before the time expires then the system will accept the pre-selected option.
4. Auto[matic] with feedback: a yes/no question is provided but only a single option is provided, which represents the action that will be taken by the system; the operator may acknowledge the selection, but may not change the decision.
5. Auto[matic]: the system chooses whether or not to authorize the weapon release.

3.2.2 Image Analysis

For the image analysis task, an operator must examine an image and answer a question about the image. This task also has a time limit, which, if it expires, indicates that the operator did not accomplish that task.

1. Manual: a question is provided about the image and the operator provides an answer.
2. Multiple Options: a question is provided along with a list of possible answers; the operator chooses an answer from the list.
3. Multiple Options with Consent: a question is provided along with a list of possible answers; one suggestion is pre-selected, which represents the system's suggestion; the operator chooses an answer from the list; if time expires before an answer is selected, the system will not take any action to identify the image.
4. Consent: a question is provided along with a single answer, which represents the system's suggestion. The operator may accept or reject this answer. If time expires before the operator acts, then no action is taken by the system to identify this image.
5. Multiple Options with Veto: a question is provided along with a list of possible answers; one suggestion is pre-selected, which represents the system's suggestion; the operator chooses an answer from the list; if time expires before an answer is selected, the system will identify the image with the pre-selected option.

6. Veto: a question is provided along with a single answer, which represents the system's suggestion. The operator may accept or reject this answer. If time expires before the operator acts, the system will identify the image with the answer provided.
7. Auto[matic] with Feedback: a question is provided along with a single option, which represents the system's answer. The operator may acknowledge the selection, but may not change the decision.
8. Auto[matic]: the system acts automatically; no feedback is provided to the operator.

3.2.3 Allocation

The allocation task involves assigning tasks to sorties. The allocation becomes active whenever a vehicle loses sensors or weapons or if a new task is created. Once activated, the operator can either re-assign old tasks (if necessary) or assign new tasks.

1. Manual: the operator manually assigns tasks to sorties and orders the task for each sortie.
2. Auto-Sequence: the operator manually assigns tasks to sorties and the system optimizes the ordering of those tasks.
3. Auto-allocate: The operator can manually allocate and invoke the system to optimize the ordering of tasks. However, the operator may also select a subset of tasks and sorties and invoke the system to automatically allocate tasks to sorties.
4. Auto[matic]: the system will automatically allocate. The operator still has a chance to modify the results, using the same tools available in lower LOAs.

3.2.4 Autorouting

Autorouting is invoked for two different reasons in ALOA. The first reason is in response to pop-up threats. In that case, the system will replan (using the existing mission task lists) to try and create more survivable routes against the new threat laydown. The second reason is in response to a change in the task allocation. In that case, the system must replan so that the aircraft have routes that can achieve their assigned tasks. The LOAs are the same for these cases, but the behavior once a route is selected is slightly different.

In response to a pop-up threat, one or more routes are generated, depending on the autonomy level, and a route is selected by the operator or the system, again, depending on the autonomy level. The selected route is flown by the sortie.

In response to an allocation change, however, each aircraft must be assigned a route before any routes are allowed to change in the system. Thus, if a sortie is rerouted during allocation, its route is approved but not committed until the other sorties have routes. This distinction is described in more detail below.

Each autorouting LOA is first discussed as it pertains to pop-up threats. Following that is a discussion of the difference in behavior that occurs during an allocation.

3.2.4.1 Autorouting in Response to a Pop-up Threat

1. Manual: the operator may drag the route to change the route plan; the system will automatically compute route metrics for the modified route and commit the modified route to the aircraft.
2. Multiple Options: the system generates one or more options. The operator may select the current route or choose another one. Once a route is selected, it is committed to the aircraft.
3. Multiple Options with Consent: the system generates one or more options, one of which is highlighted, representing the system's suggested route. The operator may select the highlighted route or choose another one. Once a route is selected, it is committed to the aircraft. If time expires before action is taken, then the current route is used.
4. Consent: the system generates a new route, which is highlighted and represents the system's suggested route. The operator may select that route or choose the current route. Once a route is selected, it is committed to the aircraft. If time expires before action is taken, then the current route is used.
5. Multiple Options with Veto: the system generates one or more options, one of which is highlighted, representing the system's suggested route. The operator may select that route or choose another one. Once a route is selected, it is committed to the aircraft. If time expires before action is taken, then the computer's suggested route is used.
6. Veto: the system generates a new route, which is highlighted and represents the system's suggested route. The operator may select that route or choose the current route. Once a route is selected, it is committed to the aircraft. If time expires before action is taken, then the computer's suggested route is used.
7. Auto[matic] with Feedback: the system automatically generates a route and commits it to the aircraft. The system provides feedback about the newly generated route to the operator.
8. Auto[matic]: the system automatically generates a route and commits it to the aircraft without any feedback to the operator.

3.2.4.2 Autorouting for Allocation

During an allocation, routes are not immediately committed to the aircraft because it is important to see the whole collection of routes before accepting the allocation. So instead of accepting and committing individual routes, each route must simply be approved. Once all routes are in the approved state, then all of the routes are committed to each of the aircraft.

For the Veto and Multiple Options with Veto autonomy levels, the system takes action only when the time expires. Thus, the options are approved but only take effect at the end of the planning cycle. When an aircraft is in the Veto and Multiple Options with Veto autonomy levels, its status will automatically become approved when the routes become available. The operator may choose to approve each of those routes, which will change the status to Approved and will cause the routes to change earlier in the planning cycle. If no action is taken, though, then the routes will be committed at the end of the planning cycle

If an aircraft is assigned new tasks during an allocation then its current route is no longer valid. Thus, it will not be possible to approve the current route as an option and the operator will not be allowed to select it.

If an aircraft is in the Auto with Feedback or Auto autonomy level, the system automatically selects the new routes. However, as mentioned earlier, all routes must be approved before any routes are committed. Therefore, during an allocation, if a sortie has the Automatic with Feedback or Automatic autonomy levels, then its status will automatically appear as Approved. Neither the current route nor the new route can be manually approved. Thus, the operator may not override the computer's decision. Once all routes are approved, then all routes are committed. In addition, in Automatic with Feedback, the system will indicate to the operator that the routes are approved.

3.3 Adapting Levels of Autonomy

As noted above, another human-centered automation notion is that the level of autonomy does not need to be fixed - it can be allowed to vary depending on the situation. When allowing autonomy levels to change, the assignment of autonomy levels will be dynamic and can vary over time depending on the situation. This approach recognizes that control must pass back and forth between the human and computer to optimize system performance.

Although the notion of adapting LOA offers a promising method for human-centered control, questions remain about how it should be implemented. A number of adaptive autonomy strategies have been proposed for invoking automation, including critical events, human performance measures, psycho-physiological assessment of operator workload, and behavior modeling. According to Kaber and Endsley, "…studies demonstrate that critical events and performance approaches to adaptive autonomy may be effective for moderating operator workload in various cognitive tasks." The authors go on to note "adaptive autonomy may provide performance benefits to operators involved in monitoring, psychomotor and dynamic control tasks. These benefits appear to result from maintaining operator involvement in active control and managing workload…" [2]

To provide AFRL tools to experiment with LOA that can be adapted, part of the ALOA research effort was to design and implement schemes for adapting the LOA. Two general methods of the adapting LOA are used in ALOA: *adaptable* autonomy, in which the operator (pilot) of the system can manually change LOA; and *adaptive* autonomy, in which the system determines when and how to change the LOA. (A third notion for adapting LOA, not currently implemented in ALOA, is *adjustable* autonomy, in which the operator sets the parameters that the system uses for adaptive autonomy.) A dialog (see the right side of Figure 4 ALOA's Right Screen) shows how the user is presented with options.

3.3.1 Adaptable Autonomy

In the adaptable method implemented in ALOA, the operator uses the controls in the dialog to change the LOA. The LOA control panel allows the operator to select the LOA for route planning (AR), image analysis (IM), weapon release authorization (WR), and allocation (AL), using the slider bars in the upper middle portion of the dialog. Alternatively, the system LOA can be increased or decreased by pressing the appropriate button. The four component LOA will

change accordingly. The current level of autonomy for each task is listed near the top of the dialog.

If enabled by the researcher, there is a “Full Auto” button at the top of the dialog. This is a system override button that sets all autonomy levels to levels defined by the researcher during experiment set up.

The LOA for route planning can be set “globally”, meaning that a single level is set for all sorties, or each sortie could have its own autonomy level. For example, if Sortie A was assign to strike a high-value target in an area where significant collateral damage could occur and the other sorties were performing routine ISR tasks, it might be desirable to have a lower LOA for Sortie A and high LOAs for the other sortie so that the operator could focus attention on the strike task. The LOA control panel allows the operator to set a global route planning LOA or to set the LOA for individual sorties.

3.3.2 Adaptive Autonomy

ORCA has implemented three schemes for adaptive autonomy: workload, performance, and time-based. An adaptive controller automatically changes the autonomy levels based on input from the adaptive scheme. When and how autonomy levels change in each of these schemes is set by the researcher during experiment set up.

Regardless of the adaptive scheme used in an experiment, the first step in setting up the adaptive autonomy used in the experiment is to fill out a table of combinations of LOA that will be used. (Without limiting the number of combinations, the default would by 4 (AL) x 8 (AR) x 8 (IM) x 5 (WR) = 1280.) Each row in the table represents a system level of autonomy. Below (Figure 2) is an example; the first column gives the system level LOA and the numbers in the columns show the task LOA.

LOA	AR	AL	WR	IM
1	1	1	1	1
2	2	2	2	1
3	2	2	3	2
4	3	3	3	3
5	4	3	3	4
6	5	3	4	5
7	5	3	4	6
8	6	4	4	6
9	7	4	5	7
10	8	4	5	8

Figure 2 System Levels of Autonomy

3.3.2.1 Workload

In the workload scheme, the researcher associates numerical workload levels for individual tasks that will occur during an experimental run. Then workload thresholds are set that represent conditions of overwork and underwork. The researcher also sets controls for how the autonomy level should be changed (up or down) when thresholds are crossed. During the experiment, current workload is computed periodically and compared against the workload thresholds. When workload crosses a threshold, the autonomy level is changed according to the rules set by the experimenter. In general, we expect that the system LOA will change by one level. As an example, if the workload is deemed to be too low to engage the operator, the autonomy level would be changed to increase the manual component.

The workload scheme can be triggered by exogenous tools for measuring workload. In the ALOA test bed, however, workload is simply estimated by the existence of tasks. It would be exciting to use physiological measurements or other tools to trigger changes in the system LOA.

3.3.2.2 Performance

In the performance scheme, the researcher sets performance measures, such as the number of seconds to clear a red plane or analyze an image, and performance thresholds that represent the operator over-achieving or under-achieving. As with the workload scheme, the researcher sets controls for how the autonomy level should be changed (up or down) when thresholds are crossed. During the experiment, a performance score is computed periodically and compared against the performance thresholds. When the performance score crosses a threshold, the autonomy level is changed according to the rules set by the experimenter. In general, we expect that the system LOA will change by one level. As an example, an underachieving operator would have the autonomy level increase.

3.3.2.3 Time-based Control

In this scheme, the researcher sets times at which the autonomy level changes. The time and new LOAs can be set in the experiment script. The researcher can use this method to accommodate autonomy level changes based on events, such as change in mission phase or the completion of a task. There are no restrictions on how and when the researcher can change the autonomy levels.

4 The ALOA Test Bed

The ALOA test bed is an instrumented ground control station emulator that presents a operator with four primary tasks and several secondary tasks that help measure workload. There are also panels and tools to enhance an operator's situation awareness. The primary tasks presented to the operator, which all have multiple levels of autonomy available, are autorouting, task allocation, image analysis, and weapon release authorization. In addition, the system has been instrumented to measure performance, which enables researchers to perform trials and produce data to support human effectiveness research.

There are two primary screens. The left screen (see Figure 3) provides access to tools for autorouting, allocation, image analysis, weapon release authorization, instantaneous vehicle position, and figures of merit for both force level and sortie level. The right screen provides a

timeline and other tools to monitor the scenario and vehicle health and status. It also displays status updates, and lets the operator monitor and manipulate levels of autonomy.

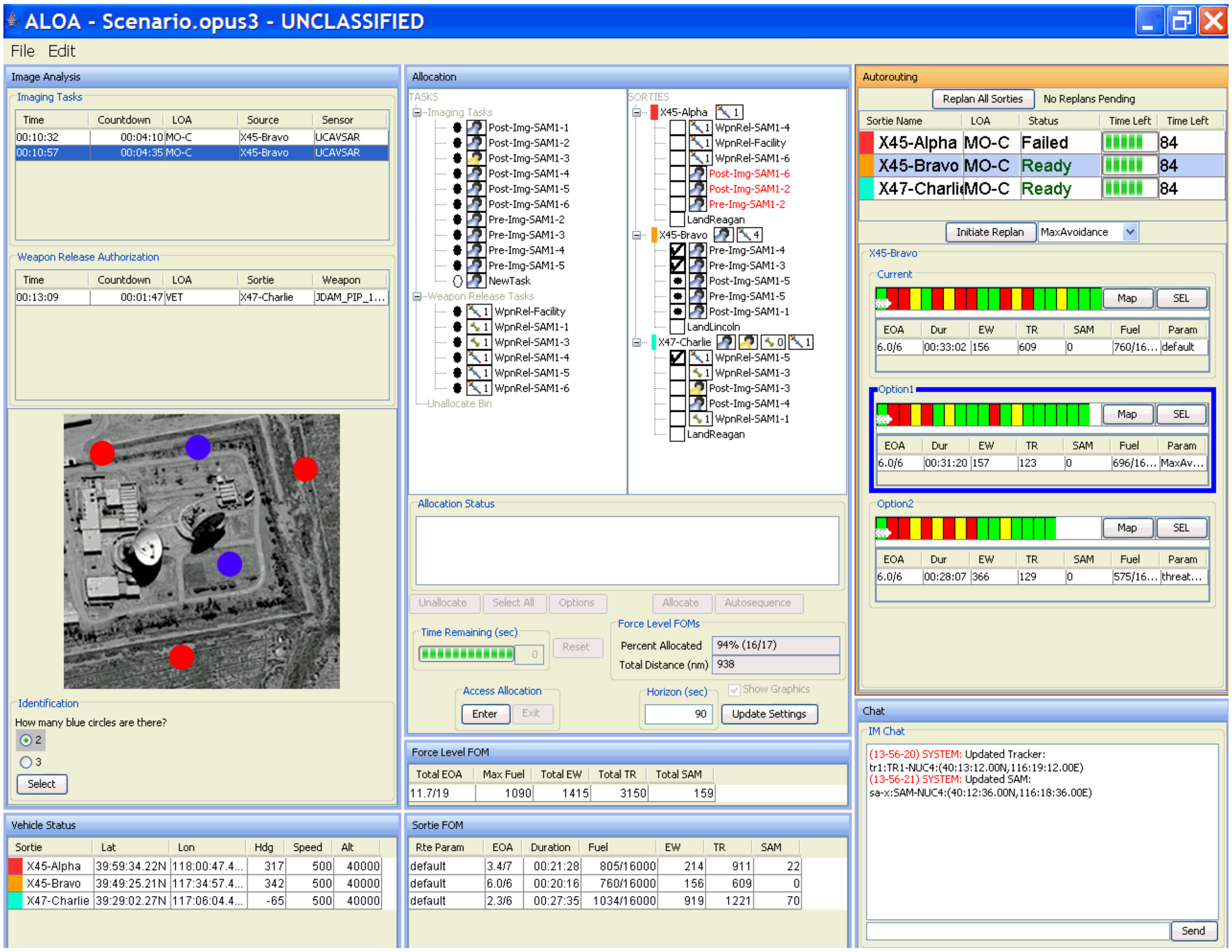

Figure 3 ALOA's Left Screen

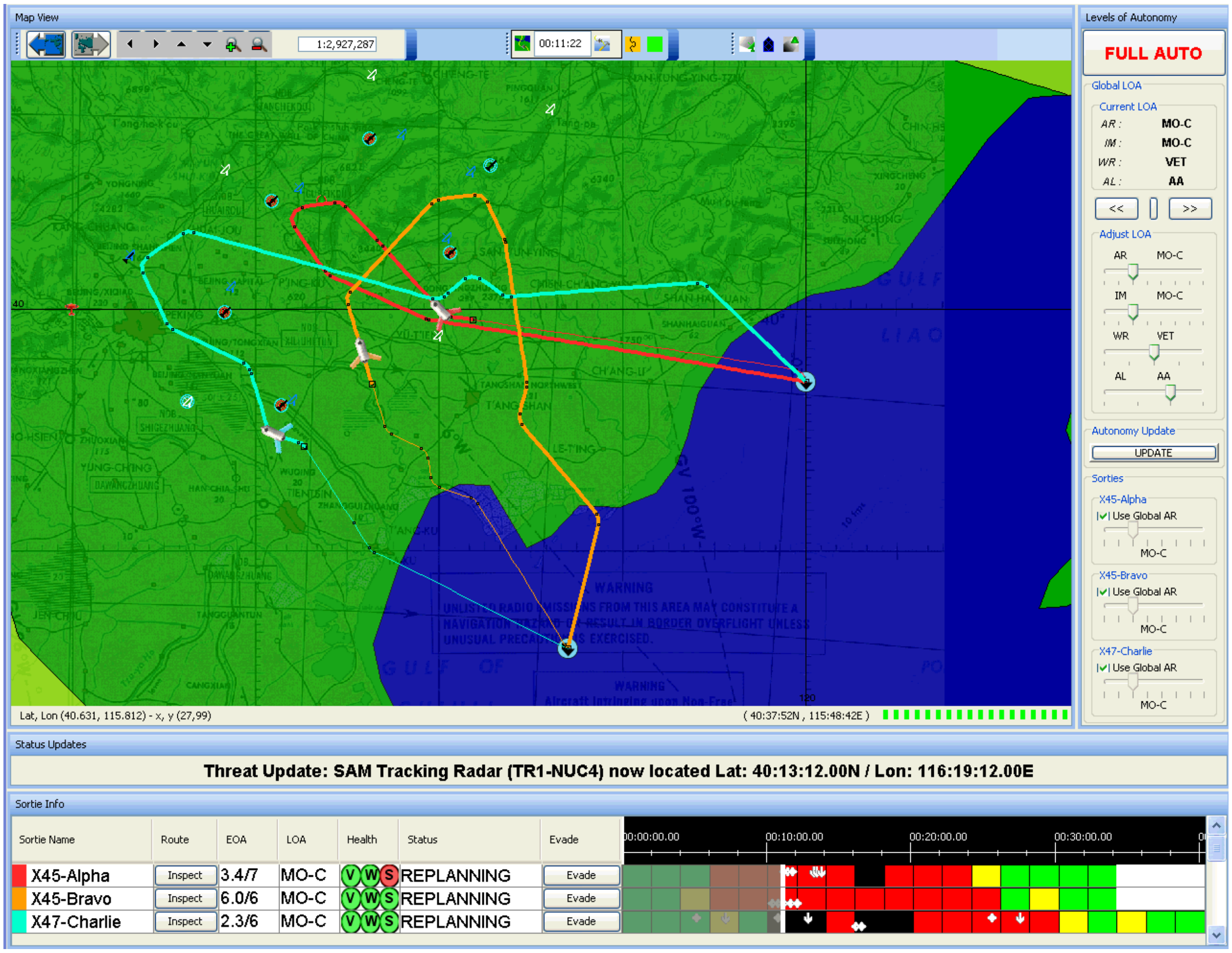

Figure 4 ALOA's Right Screen

4.1 Situation Awareness

ALOA provides a map (see Figure 4) that can display routes, threats, task positions, keepout zones, and other elements of the area of responsibility using user defined colors, line styles, and icons. CADRG and DTED data can also be displayed as semi-transparent overlays for enhanced situation awareness. A timeline is displayed for each sortie that is under the operator's control. The timeline depicts when tasks will occur in time and highlights which portions of the route are more dangerous than others. The timeline is interactive and when a section of the timeline is pressed the corresponding section of the route is displayed on the map.

Health and status, instantaneous position, and figures of merit are displayed for each sortie. Figures of merit indicate data such as probability of survival, expected number of tasks achieved, fuel consumed, and amount of threat exposure. Force level figures of merit are also displayed, which are aggregate values and provide an indication of how well the UAVs are performing.

4.2 Autorouting

ALOA accesses OPUS mission planning tools to automatically generate routes given the user-specified mission constraints. The OPUS autorouter produces goal-seeking, threat-avoiding,

terrain-aware routes that consider a number of constraints such as vehicle performance and wind. Eight levels of automation have been provided.

Routes that are produced in ALOA are based on a named set of routing parameters. The researcher may create rules of engagement (ROEs) that dictate what routing parameter set to use at different times in the scenario. The operator must then be aware of the routing parameters and generate routes appropriately. Each route option that is produced is accompanied by its own timeline depicting danger areas and task completion times as well as figures of merit. The operator may view a route option on the map and finally select a route to be flown.

4.3 Allocation

A typical approach to solving allocation problems is to divide the problem into its two distinct aspects – assignment and sequencing. The assignment problem addresses which asset should get which task. Since not all assets have the resources to accomplish every task, it important to be able to quickly assess what assets are capable of what tasks. Sequencing determines the order in which tasks are performed. The answer to the sequencing problem is called a ***tieup***. Physical locations of assets and tasks are important in being able to create efficient and feasible task sequences. The sequencing problem is sometimes known as the Traveling Salesman Problem. Additional complications arise when time constraints and cooperation require additional coordination between how tasks are assigned and how they are sequenced.

The allocation panel provides situation awareness into task progress and sortie assignments as well as provides access to OPUS mission planning tools, which can automatically allocate tasks to sorties. All tasks and their assignments are displayed in trees. As tasks are completed they are checked off. At any point the operator may manually drag an uncompleted task from one sortie to another, or move it to a different position in the same sortie's mission task list (tieup). The tieup can be displayed on the map as a dotted line between each task location. The tieup can also be edited directly on the map. The operator can choose to autosequence a sortie's tieup, which accesses OPUS tools and generates an optimal ordering of the tasks currently assigned to that sortie. Alternatively, the operator can select any subset of tasks for reallocation and select a subset of sorties and access OPUS tools to automatically allocate the tasks to sorties. Once tieups are generated, the operator can access the autorouting tools to generate routes that use the new mission task lists. Allocation provides four levels of autonomy including fully automatic, auto-allocate, auto-sequence, and manual.

The effectiveness of an allocation is not known until detailed routes are generated although the allocation panel does provide immediate nominal feedback to the operator whenever changes are made. For example, if a task cannot be completed then the task name is colored in red. Also, the straight line tieup distance is computed, which gives a nominal estimate on route length.

4.4 Image Analysis

As imaging tasks are completed by an aircraft, imagery becomes available in a table. The operator may select an image at any point and attempt to annotate or identify the image. The images, the question about an image, any suggested answers, and the correct answer are all defined by the researcher in a configuration file. Eight levels of autonomy have been developed for image analysis that all rely on an unrealistic automatic target recognition capability, which

the researcher must define through the configuration file. The images can expire, which is also researcher-controlled, so the task must be performed in a specific time interval.

4.5 Weapon Release Authorization

Before each weapon release an image will appear, which the operator must analyze to determine whether to authorize the release or not. The image will appear in a table a period of time before each weapon release, which is researcher-controlled. The question posed to the operator will be a yes/no question, which corresponds to whether the release should be authorized or not. Five levels of autonomy have been developed for weapon release authorization that all rely on a fictitious automatic target recognition capability, which the researcher must define through the configuration file.

4.6 Secondary Tasks

ALOA has secondary tasks that a researcher can employ to help measure workload:

- **Health and Status** – Circles that represent the status of a vehicle's communication capability, weapons, and sensors are displayed. See Figure 5. The researcher may create script events to change the status from green to yellow and from yellow to red. If the status becomes red then that capability is lost for the remainder of the scenario. If the status becomes yellow then the operator must press the status button to reset it to green. The researcher can therefore monitor how long the operator takes to recognize changes in the vehicle health and status.

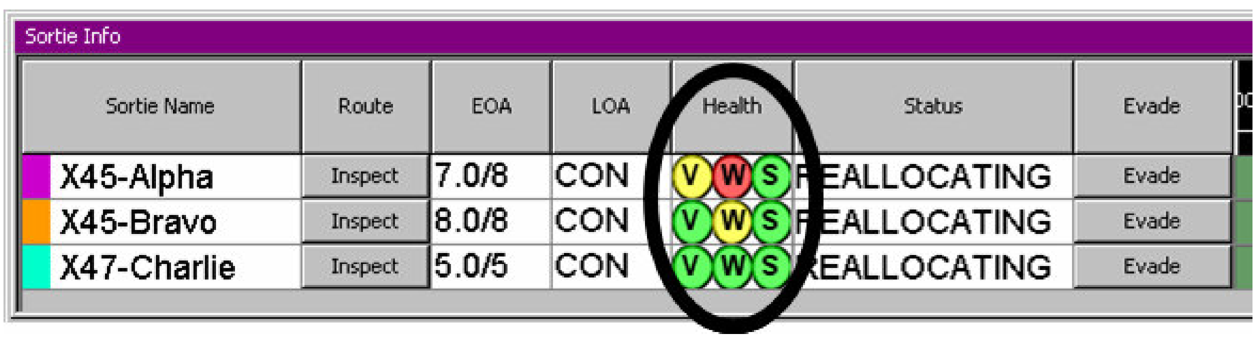

Figure 5 Sortie Health and Status Bar

- **Red Plane** – The researcher may script an event to place a red plane icon, which is customizable, on the map. The operator must then press the red plane icon and enter a sequence of characters, which can also be configured. See Figure 6. The researcher can monitor how long the operator takes to recognize the red plane and clear it.

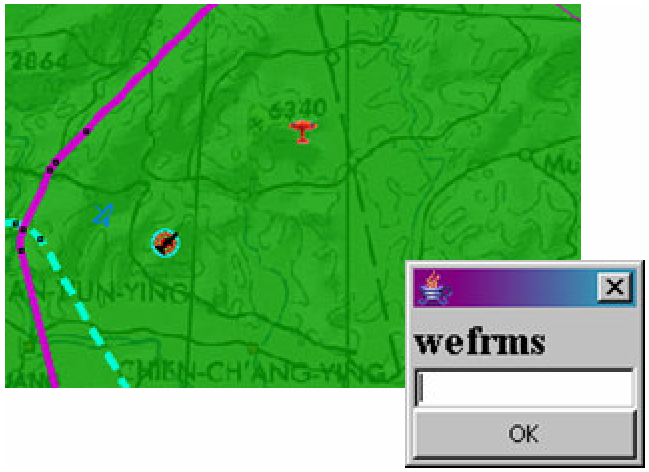

Figure 6 IFF - Red Plane

- **SAM Shots** – If the route passes through SAM exposure then there is a possibility of a SAM shot. If a SAM shot occurs then the operator must quickly press the evade button. This test can help measure whether the operator was aware that an aircraft was passing through SAM exposure.
- **Chat Display** – The researcher may script events to enter text in the chat display. Some of the text events may require a response from the operator.

4.7 Autonomy Adaptability

ALOA provides a panel that enables the operator to change the autonomy level for autorouting, allocation, image analysis, and weapon release authorization. In addition, a "Full Auto" button is available, which the operator may press at any time if the tasks become too overwhelming. This causes the system to essentially take over control until the operator is comfortable to resume.

ALOA also provides three techniques that enable the system to automatically adapt the levels of autonomy. The techniques include workload-based, performance-based, and time-based. The researcher may specify parameters for these techniques depending on the type of scenario. As the system monitors the scenario it will automatically increase or decrease the levels of autonomy depending on the adaptive technique employed.

4.8 Script and Configuration Editor

ALOA provides the researcher with a script editor to help develop scenarios. Script events are executed at researcher-defined times during the scenario. Possible script events in ALOA include the ability to pop-up, move, or delete threats, create new tasks, change the vehicle health and status, insert red planes, set chat text, execute a Situation Awareness Global Assessment Technique (SAGAT) survey, adjust the simulation pace, adjust levels of autonomy, set autorouting parameters, blank or end the simulation, adjust the lookahead time for replans, play sounds, and perform a screen capture. These scripts may be saved to a file and executed with a given scenario.

ALOA also provides a configuration editor, which helps define parameters in a scenario. The parameters include associating images, questions, suggestions, and answers with targets, how many options to generate when autorouting, whether to show confirmation dialogs, how long to show threat circles and blink routes, how many characters to display in a red plane test, and whether to include a Full Auto button.

4.9 Architecture

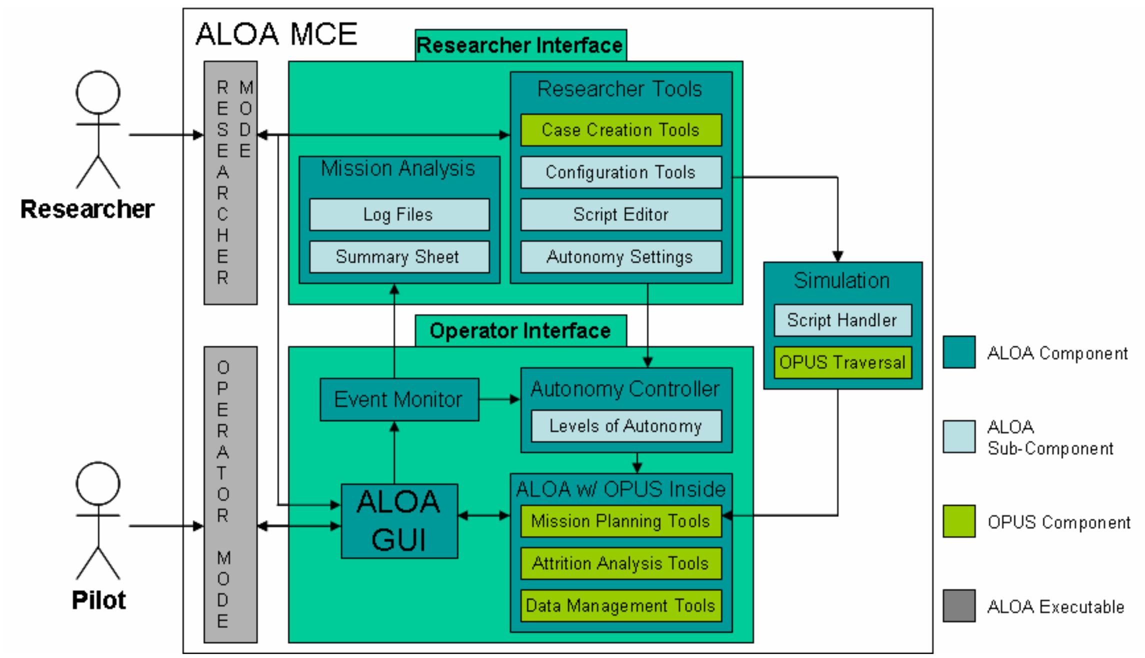

Figure 7 ALOA2 Architecture

ALOA has interfaces and tools for both the researcher and the operator. See Figure 7. The researcher interface provides tools to set up the experiment and manipulate the simulation. The operator interface provides tools to monitor the scenario and access OPUS, which provides mission planning, analysis, and data management capabilities. The system was designed to emulate a mission control element.

5 Future Directions for Research

5.1 Chat Display

A chat interface can be an effective secondary tasking tool for measuring workload. As Missy Cummings discusses in [3], the "use of the embedded chat tool to induce information-seeking secondary tasks yielded critical results needed for determination of operator workload." Although ALOA provides several embedded secondary tasks, chat provides the researcher with another important tool for measuring workload. The chat display, however, must be designed in

a way to make information retrieval easy. The operator should not need to spend much time with the chat display because it would distract from the primary tasks.

The chat display in ALOA currently shows all chat sessions in one area. There are several improvements that could be applied to the interface. For example, changes to the rules of engagement could be in one tab, changes to the environment could be in another tab, and status questions could be asked in another tab. It is possible to search the chat or manually group data together. Other possibilities to improve the chat display could be explored with future research.

5.2 Additional Adaptive Schemes

ORCA implemented three adaptive schemes in Phase II: workload-based, performance-based, and time-based. In future research, several more schemes could be explored.

5.2.1 Phase of Mission

The geographic location of the aircraft as it heads into and out of the area of responsibility (AOR) could be a driver for adaptive changes in autonomy. For example, aircraft entering the AOR, inside the AOR, and exiting the AOR could all be at different LOAs. Aircraft heading into the AOR must react to changes in the environment, such as pop-up tasks and threats, however, those changes typically affect the route in the future. Thus, the operator has a relatively large amount of time to react. Aircraft in the AOR, however, may need to react immediately to changes in the environment. Aircraft leaving the AOR are finished with their mission and have to worry less about changes in the environment. In addition to the AOR, there may be other geographic-based reasons to adapt the LOAs as well. For example, no-fire zones, deconfliction zones, or enemy-controlled strongholds may be reasons to adapt. The system could be made aware of these geographic areas and adapt the LOAs appropriately.

5.2.2 Future Workload

The workload-based scheme that ORCA developed in Phase II is based on the current workload. In that scheme, the system adapts in real-time as new tasks require operator attention. However, it is possible to predict future workload and move the system into a different LOA before those events occur. For example, image analysis and weapon release authorization events are known once the routes are produced. The system could analyze those events and if many events occur in close proximity then the system may introduce an adaptive change in LOA before the events occur. The system is also aware of future SAM exposure, which may result in SAM shots, and could adapt the LOA during times that require heightened awareness. Also, if vehicle health and status monitoring is required at regular intervals then the system can factor that secondary task into future workload requirements. Adaptive changes to LOA that are known ahead of time could be displayed on the timeline to provide additional situation awareness to the operator.

5.3 ALOA Route Server

Autorouting and allocation take time and resources to produce solutions during a mission. When all processing is done on the same computer then more aircraft, and therefore more autorouting, will require longer overall replan times. For example, if a system is in an automatic LOA, which implies that routes are committed as soon as they are ready, then all aircraft after the first one must wait before they can have a route committed. It is still an issue even when the operator has

to consent to routes. Since the routes must be generated sequentially, when using a single computer, then an order must be used when autorouting the aircraft. However, the operator may want to spend more time analyzing one route versus another and would prefer the route that requires more attention to be generated first. Unfortunately, the operator may not know until the routes are returned which ones require more attention. Thus, requiring the operator to specify an order is not practical.

A future research direction could be to build a route server that generates routes. It would then be possible to have a route server for each aircraft that is being controlled by the operator. Then, all routes would be available as quickly as possible. By placing the replanning process on a different computer then the control station still maintains full control of the resources of its computer, which will also make the user interface more responsive.

5.4 Additional Allocation Constraints

The allocation component of ALOA currently treats all tasks as independent and of equal value. That component could be enhanced to handle synchronization and timing constraints, resource and asset value, and task priority. If timing constraints are handled during the allocation then the imaging tasks in ALOA could serve as the weapon release authorization. The timing constraint would ensure that the imaging task occurs some minimum time before the weapon release. If resources or assets can have different values then that would impact task assignments because lower value assets could presumably be given more dangerous missions. If task priority is handled during the allocation then the operator may be told new rules of engagement such as to handle any and all tasks above a certain priority but no others.

The operator must be able to monitor all of these constraints. However, if the LOA is manual and requires operator intervention then the user interface must allow for control as well. Presenting user interfaces to visualize these constraints is a future research direction. There is a multitude of information that needs to be visualized in many places such as on the map and on a timeline.

5.5 Mission Management

ALOA currently requires the researcher to specify whether a pop-up threat should invoke a system replan. However, if a system replan is invoked then all sorties are rerouted. Often, a pop-up threat will only affect a subset of the routes under a operator's control. A mission management component would be responsible for analyzing changes to the environment and invoking the autorouter when appropriate. This would reduce the workload by not overloading the operator with route options that do not need to be considered.

ALOA also requires the researcher to specify which route option is correct. The mission management component could provide logic to assist the researcher in this task. It should be noted that route planning and allocation problems are notoriously difficult to solve optimally so that no known automated tools can guarantee the best answers.

5.6 Sandbox Visualization

ALOA provides some sandbox visualizations but much more can be realized. During replans, the operator can now visualize proposed changes by viewing a dashed-line version of the routes and tieups on the map. There is also a traversal tool that lets the operator project a sortie's position into the future. However, that tool simply projects the aircraft's icon into the future. This

prevents the operator from visualizing the aircraft's current position. One effort may be to create a transparent aircraft icon that is projected into the future while the current aircraft's icon remains solid.

There are many other tasks that could be performed in a sandbox, however. For example, the operator may want to perform what-if studies to determine how a pop-up threat might affect a sortie's route. The operator may want to make these what-if decisions when choosing between new routes during a replan or may want to analyze their current route for weaknesses. The operator could also receive chat messages that ask questions such as how vulnerable are your aircraft to a certain area. The system may want to automatically adapt into different levels of autonomy while the operator is performing these what-if studies because the operator's attention will be diverted.

5.7 Interactive Script Development

Developing a script as the researcher requires that script events are placed at certain times. The researcher may currently run a mission and input script events in the future but it is not possible to go backwards without starting over from the beginning. A future research direction is to reduce the time required by the researcher to develop a script. This would involve being able to jump to particular times in the scenario (i.e. resetting and executing all events up to that point as quickly as possible). The researcher could also more easily try an event and undo the changes if that event is not desired. There could also be a mechanism to insert an event at the current time. Currently, the researcher has to manually set the event time appropriately.

The researcher would also be able to create script events graphically through the map. For example, dragging a threat would populate an event for the script editor. Currently, the researcher must choose a location on the map and edit a separate dialog in the script editor for its location. This improvement would simplify the task of the researcher during script development.

6 Summary

ORCA has completed the design and implementation of a human factors test bed for implementing and evaluating a range of adaptive levels of autonomy for UAV supervisory control. Preliminary versions of the test bed were used in experiments conducted by AFRL. The software delivered at the end of Phase II is a result of a spiral design approach that allowed significant input from our Air Force customer, both in the early design phase and after testing and experimenting with the preliminary versions of the software. In short, the goals for Phase II have been met.

Like many long research efforts, there was an evolution in some goals as it became clear what was most useful and what was not, including changes in the numbers of levels of autonomy for the various operator tasks, and schemes for implementing adaptive autonomy. In addition, more work was done on the researcher end, including designing tools to streamline the scenario and script generation processes for setting up experiments.

Phase II work has built a foundation for further ALOA test bed design and enhancement, including

- enhancement to the chat window,
- additional adaptive autonomy schemes,

- the addition of a route server to the architecture to reduce demands on the primary ALOA machine,
- additional allocation constraints including synchronization, timing, and task priority levels,
- a sandbox visualization feature to allow the operator to play out proposed mission changes and to perform "what-if" analysis, and
- an interactive script development capability to aid the researcher in experimental set up.

While these ideas would provide desirable functionality to ALOA, the test bed designed and delivered in Phase II to AFRL represents a leap forward in capability for human factors experimentation. This test bed can be made available to other researchers who need a robust environment to examine issues related to UAV supervisory control.

References

1. Parasuraman, R., Sheridan, T. B., Wickens, C. D. (Cognitive Science Lab, Catholic University of America), A Model for Types and Levels of Human Interaction with Automation, *IEEE Transactions on Systems, Man, and Cybernetics- Part A: Systems and Humans*, Vol. 30, Issue 3, pp. 286-297, May 3, 2000.
2. Kaber, B. K. and Endsley, M.R. (North Carolina State University-SA Technologies), The Effects of Levels of Automation and Adaptive Automation on Human Performance, Situation Awareness and Workload in a Dynamic Control Task, *Theoretical Issues in Ergonomics Science,* 5(2), pp. 113-153, 2004.
3. Cummings, M. L. (Massachusetts Institute of Technology) and Guerlain, Stephanie (University of Virginia), Using a Chat Interface as an Embedded Secondary Tasking Tool, *2nd Annual Human Performance, Situation Awareness, and Automation Conference*, March 2004.

Bibliography

1. Endsley, M. R., (Texas Tech University), Automation and Situation Awareness, *Automation and Human Performance: Theory and Applications*, 1996, pp. 163 - 181, Mahwah, NJ: Lawrence Erlbaum.

2. Computers That Read Your Mind, *The Economist Technology Quarterly*, September 23, 2006.

3. Bell, B. (CHI Systems, Inc.), Clark, J. (Lockheed Martin Aeronautics Company), Bringing Command and Control Down to Earth, *The 21st Digital Avionics Systems Conference,* Irvine, CA, Vol. 2, pp. 7E1-1 – 7E1-10, October 27-31, 2002.

4. Tso, K. , Tharp, G.K., Tai, A.T. (IA Tech., Inc.), Draper, M.H., Calhoun, G.L., Ruff, H.A.(AFRL/HECP, WPAFB, OH), A Human Factors Testbed for Command and Control of Unmanned Air Vehicles, *The 22nd Digital Avionics Systems Conference,* Indianapolis, IN, Vol. 2, pp. 8.C.1 - 81-12, October 12-16, 2003.

5. Kaber, B. K. and Endsley, M.R. (North Carolina State University-SA Technologies), Level of Automation Effects on Performance, Situation Awareness, and Workload in a Dynamic Control Task, *Ergonomics*, Vol. 42, No. 3, pp. 462-492, 1999.

6. Johnson, R., Leen, M., Goldberg, D., Chiu, M. (OR Concepts Applied), Adaptive Levels of Autonomy (ALOA) for UAV Supervisory Control, Final Report for May 2004 to February 2005, AFRL-HE-WP-TR-2005-0062, Air Force Research Laboratory, Wright-Patterson AFB, OH, May 2005.

7. Moffitt, V. Z., Franke, J.L., Lomas, M. (Lockheed Martin Advanced Technology Laboratories), Mixed-Initiative Adjustable Autonomy in Multi-Vehicle Operations, *Proceedings of the AUVSI's Unmanned Systems North America 2006,* Orlando, FL, August 29-31, 2006.

8. Kay, G. (Georgetown University and Washington Neuropsychological Institute), Dolgin, D., Wasel, B., Langelier, M. (Naval Air Warfare Center, Aircraft Division, Crew Systems), Hoffman, C. (Hoffmann Research Associates), Identification of the Cognitive, Psychomotor, and Psychosocial Skill Demands of Uninhabited Combat Aerial Vehicles (UCAV) Operators, Patuxent River, MD, June 21, 1999.

9. Ruff, H. A. (Sytronics, Inc.), Narayanan, S. (Wright State University), Draper, M. H. (AFRL), Human Interaction with Levels of Automation and Decision Fidelity in the Supervisory Control of Multiple Simulated Unmanned Air Vehicles, *Presence: Teleoperators and Virtual Environments (1054-7460)*, Vol. 11, No. 4, pp. 335-351, August 2002.

10. Clough, B.T. (Air Force Research Laboratory: Wright-Patterson AFB), Metrics, Schmetrics! How The Heck Do You Determine A UAV's Autonomy Anyway?, *AIAA's 1st Technical Conference and Workshop on UAV's,* AIAA-2002-3499 Portsmouth, VA, May 20-23, 2002.

11. McCarley, J. S. and Wickens, C. D. (Institute of Aviation-University of Illinois), Human Factors Implications of UAVs in the National Airspace, Technical Report AHFD-05-05/FAA-05-01, April 2005.

Table of Acronyms

AFRL	Air Force Research Laboratory
ALOA	Adaptive Levels of Autonomy
AOR	Area of Responsibility
ATC	Automatic Target Cueing
ATR	Automatic Target Recognition
AUVSI	Association for Unmanned Vehicle Systems International
BDA	Battle Damage Assessment
C2	Command & Control
CADRG	Compressed Arc Digitized Raster Graphics
DMPI	Designated Mean Point of Impact
DTED	Digital Terrain Elevation Data
FOM	Figures of Merit
HSI	Human-System Interface
ISR	Intelligence, Surveillance, and Reconnaissance
J-UCAS	Joint Unmanned Combat Air System
LOA	Level of Autonomy
MCE	Mission Control Element
MCSE	Mission Control Station Emulator
OPUS	ORCA Planning and Utility System
ORCA	OR Concepts Applied
ROE	Rules of Engagement
SAGAT	Situation Awareness Global Assessment Technique
SAM	Surface-to-Air Missile
SBIR	Small Business Innovation Research
TCS	Tactical Control System
UAV	Unmanned Air Vehicle
UCAV	Unmanned Combat Air Vehicle

Table of Abbreviations

AL	Allocation
AR	Autorouting (Automatic Route Planning)
IM	Image Analysis
WR	Weapon Release Authorization